jc916 U.S. PTO
09/15/00

09-18-00

A

PATENT APPLICATION
Express Mail Label No. *EL610288115US*
Attorney Docket No. SUN-P4431-ARG

jc920 U.S. PTO
09/662553
09/15/00

IN THE UNITED STATES PATENT AND TRADEMARK OFFICE

UTILITY PATENT APPLICATION TRANSMITTAL LETTER

Asst. Commissioner for Patents
Box Patent Application
Washington, D.C. 20231

Sir:

Enclosed for filing is an [X] original patent application or, [ ] a continuation-in-part patent application, by inventor(s) Skef F. Iterum, Declan J. Murphy, entitled METHOD AND APPARATUS FOR REACHING AGREEMENT BETWEEN NODES IN A DISTRIBUTED SYSTEM.

09662553.091500

No. of pages in Application: 23 ; No. of Claims: 43 .

No. of Sheets of Drawings: Formal: 5 , Informal: 0 .

Also enclosed are:

[ ] a claim for foreign priority under 35 U.S.C. §§ 119 and/or 365 in
[ ] a separate document [ ] the declaration;

[ ] a certified copy of the priority document;

[ ] an Associate Power of Attorney;

[ ] ___ verified statement(s) claiming small entity status;

[x] a Combined Declaration and Power of Attorney of the inventors(s);

[ ] a signed Combined Declaration and Power of Attorney of the inventors will follow;

[x] an Assignment document and form PTO-1595;

[x] a Power of Attorney by Assignee; and

[ ] Information Disclosure Statement and Form PTO-1449.

The fee has been calculated as follows:

CLAIMS					
	NO. OF CLAIMS		EXTRA CLAIMS	RATE	FEE
Basic Application Fee					$690.00
Total Claims	43	MINUS 20 =	23	$18.00=	$414.00
Independent Claims	4	MINUS 3 =	1	$78.00=	$78.00
If multiple dependent claims are presented, add $260.00					0
Total Application Fee					$1182.00
If verified statement claiming small entity status is enclosed, subtract 50% of Total Application Fee					
Add Recording Fee of $40.00 if Assignment document is enclosed					$40.00
TOTAL APPLICATION FEE DUE					$1222.00

[X] A check in the amount of $ 1222.00 is enclosed.

[ ] Application fee will follow with missing parts.

[X] Please deduct any underpayments or credit any overpayments to Deposit Account Number 50-1003.

Please direct all correspondence concerning the above-identified application to the following address:

A. Richard Park
Park & Vaughan LLP
508 Second Street, Suite 201
Davis, CA 95616
(530) 759-1661

22835
PATENT TRADEMARK OFFICE

Respectfully submitted,

By

A. Richard Park
Registration No. 41,241

Date: September 15, 2000

"Express Mail" Mailing Label No. **EL610288115US**

PATENT APPLICATION
ATTORNEY DOCKET NO. SUN-P4431-ARG

METHOD AND APPARATUS FOR REACHING AGREEMENT BETWEEN NODES IN A DISTRIBUTED SYSTEM

Inventor(s): Skef F. Iterum and Declan J. Murphy

Related Application

This application hereby claims priority under 35 U.S.C. § 119 to U. S. Provisional Patent Application No. 60/160,992 filed on October 21, 1999, entitled "Distributed Multi-Tier Mechanism for Agreement."

BACKGROUND

Field of the Invention

The present invention relates to coordinating activities between nodes in a distributed computing system. More specifically, the present invention relates to a method and an apparatus for reaching agreement between nodes in the distributed computing system regarding a node to function as a primary provider for a service.

Related Art

As computer networks are increasingly used to link computer systems together, distributed computing systems have been developed to control interactions between computer systems. Some distributed computing systems allow client computer systems to access resources on server computer systems. For example, a client computer system may be able to access information contained in a database on a server computer system.

When a server computer system fails, it is desirable for the distributed computing system to automatically recover from this failure. Distributed computer systems possessing an ability to recover from such server failures are referred to as "highly available systems."

For a highly available system to function properly, the highly available system must be able to detect a server failure and reconfigure itself so that accesses to a failed server are redirected to a backup secondary server.

One problem in designing such a highly available system is that some distributed computing system functions must be centralized in order to operate efficiently. For example, it is desirable to centralize an arbiter that keeps track of where primary and secondary copies of a server are located in a distributed computing system. However, a node that hosts such a centralized arbiter may itself fail. Hence, it is necessary to provide a mechanism to select a new node to host the centralized arbiter.

Moreover, this selection mechanism must operate in a distributed fashion because, for the reasons stated above, no centralized mechanism is certain to continue functioning. Furthermore, it is necessary for the node selection process to operate so that the nodes that remain functioning in the distributed computing system agree on the same node to host the centralized arbiter. For efficiency reasons, it is also desirable for the node selection mechanism not to move the centralized arbiter unless it is necessary to do so.

Hence, what is needed is a method and an apparatus that operates in a distributed manner to select a node to host a primary server for a service.

SUMMARY

One embodiment of the present invention provides a system for selecting a node to host a primary server for a service from a plurality of nodes in a distributed computing system. The system operates by receiving an indication that a state of the distributed computing system has changed. In response to this indication, the system determines if there is already a node hosting the primary server for the service. If not, the system selects a node to host the primary server using the assumption that a given node from the plurality of nodes in the distributed computing system hosts the primary server. The system then communicates rank information between the given node and other nodes in the distributed computing system, wherein each node in the distributed computing system has a unique rank with respect to the other nodes in the distributed computing system. The system next compares the rank of the given node with the rank of the other nodes in the distributed computing system. If one of the other nodes has a higher rank than the given node, the system disqualifies the given node from hosting the primary server.

In one embodiment of the present invention, if there exists a node to host the primary server, the system allows the node that hosts the primary server to communicate with other nodes in the distributed computing system in order to disqualify the other nodes from hosting the primary server.

In one embodiment of the present invention, the system maintains a candidate variable in the given node identifying a candidate node to host the

09552600.041500

primary server. In a variation on this embodiment, the system initially sets the candidate variable to identify the given node.

In one embodiment of the present invention, after a new node has been selected to host the primary server, if the new node is different from a previous node that hosted the primary server, the system maps connections for the service to the new node. In a variation on this embodiment, the system also configures the new node to host the primary server for the service.

In one embodiment of the present invention, the system restarts the service if the service was interrupted as a result of the change in state of the distributed computing system.

In one embodiment of the present invention, the given node in the distributed computing system can act as one of: a host for the primary server for the service; a host for a secondary server for the service, wherein the secondary server periodically receives checkpointing information from the primary server; or a spare for the primary server, wherein the spare does not receive checkpointing information from the primary server.

In one embodiment of the present invention, upon initial startup of the service, the system selects a highest ranking spare to host the primary server for the service.

In one embodiment of the present invention, the system allows the primary server to configure spares in the distributed computing system to host secondary servers for the service.

In one embodiment of the present invention, comparing the rank of the given node with the rank of the other nodes in the distributed computing system involves considering a host for a secondary server to have a higher rank than a spare.

09955255.041505

In one embodiment of the present invention, after disqualifying the given node from hosting the primary server, the system ceases to communicate rank information between the given node and the other nodes in the distributed computing system.

BRIEF DESCRIPTION OF THE FIGURES

FIG. 1 illustrates a distributed computing system in accordance with an embodiment of the present invention.

FIG. 2 illustrates how highly available services are controlled within a distributed computing system in accordance with an embodiment of the present invention.

FIG. 3 illustrates how a replica managers controls highly available services in accordance with an embodiment of the present invention.

FIG. 4 is a flow chart illustrating the process of selecting and configuring a new primary server in accordance with an embodiment of the present invention.

FIG. 5 is a flow chart illustrating some of the operations performed by a primary server in accordance with an embodiment of the present invention.

FIG. 6 illustrates how a node is selected to host a primary server through a disqualification process in accordance with an embodiment of the present invention.

FIG. 7 illustrates how nodes a disqualified in accordance with an embodiment of the present invention.

DETAILED DESCRIPTION

The following description is presented to enable any person skilled in the art to make and use the invention, and is provided in the context of a particular application and its requirements. Various modifications to the disclosed embodiments will be readily apparent to those skilled in the art, and the general principles defined herein may be applied to other embodiments and applications without departing from the spirit and scope of the present invention. Thus, the present invention is not intended to be limited to the embodiments shown, but is to be accorded the widest scope consistent with the principles and features disclosed herein.

The data structures and code described in this detailed description are typically stored on a computer readable storage medium, which may be any device or medium that can store code and/or data for use by a computer system. This includes, but is not limited to, magnetic and optical storage devices such as disk drives, magnetic tape, CDs (compact discs) and DVDs (digital versatile discs or digital video discs), and computer instruction signals embodied in a transmission medium (with or without a carrier wave upon which the signals are modulated). For example, the transmission medium may include a communications network, such as the Internet.

09552553.091500

Distributed Computing System

FIG. 1 illustrates a distributed computing system 100 in accordance with an embodiment of the present invention. Distributed computing system 100 includes a number of computing nodes 102-105, which are coupled together through a network 110.

Network 110 can include any type of wire or wireless communication channel capable of coupling together computing nodes. This includes, but is not limited to, a local area network, a wide area network, or a combination of networks. In one embodiment of the present invention, network 110 includes the Internet. In another embodiment of the present invention, network 110 is a local high speed network that enables distributed computing system 100 to function as a clustered computing system (hereinafter referred to as a "cluster").

Nodes 102-105 can generally include any type of computer system, including, but not limited to, a computer system based on a microprocessor, a mainframe computer, a digital signal processor, a personal organizer, a device controller, and a computational engine within an appliance.

Nodes 102-105 also host servers, which include a mechanism for servicing requests from a client for computational and/or data storage resources. More specifically, node 102 hosts primary server 106, which services requests from clients (not shown) for a service involving computational and/or data storage resources.

Nodes 103-104 host secondary servers 107-108, respectively, for the same service. These secondary servers act as backup servers for primary server 106. To this end, secondary servers 107-108 receive periodic checkpoints 120-121 from primary server 106. These periodic checkpoints enable secondary servers 107-108 to maintain consistent state with primary server 106. This makes it possible for one of secondary servers 107-108 to take over for primary server 106 if primary server 106 fails.

Node 105 can serve as a spare node to host the service provided by primary server 106. Hence, node 105 can be configured to host a secondary server with respect to a service provided by primary server 106. Alternatively, if all primary servers and secondary servers for the service fail, node 105 can be configured to host a new primary server for the service.

Also note that nodes 102-105 contain distributed selection mechanisms 132-135, respectively. Distributed selection mechanisms 132-135 communicate with each other to select a new node to host primary server 106 when node 102 fails or otherwise becomes unavailable. This process is described in more detail below with reference to FIGs. 2-6.

Controlling Highly Available Services

FIG. 2 illustrates how highly available services 202-205 are controlled within distributed computing system 100 in accordance with an embodiment of the present invention. Note that highly available services 202-205 continue to operate even if individual nodes of distributed computing system 100 fail.

Highly available services 202-205 operate under control of replica manager 206. Referring to FIG. 3, for each service, replica manager 206 keeps a record of which nodes in distributed computing system 100 function as primary servers, and which nodes function as secondary servers. For example, in FIG. 3 replica manager 206 keeps track of highly available services 202-205. The primary server for service 202 is node 103, and the secondary servers are nodes 104 and 105. The primary server for service 203 is node 104, and the secondary servers are nodes 103 and 105. The primary server for service 204 is node 102, and the secondary servers are nodes 104-105. The primary server for service 205 is node 103, and the secondary servers are nodes 102, 104 and 105.

Replica manager 206 additionally performs a number of related functions, such as configuring a node to host a primary (which may involve demoting a current host for the primary to host a secondary). Replica manager 206 may additionally perform other functions, such as: adding a service; removing providers for a service; registering providers for a service; removing a service; handling provider failures; bringing up new providers for a service; and handling dependencies between services (which may involve ensuring that primaries for dependent services are co-located on the same node).

Referring back to FIG. 2, replica manager 206 is itself a highly available service that operates under control of replica manager manager (RMM) 208. Note that RMM 208 is not managed by a higher level service.

As illustrated in FIG. 2, RMM 208 communicates with cluster membership monitor (CMM) 210. CMM 210 monitors cluster membership and alerts RMM 208 if any changes in the cluster membership occur.

CMM 210 uses transport layer 212 to exchange messages between nodes 102-105.

Process of Selecting a New Primary

FIG. 4 is a flow chart illustrating the process of selecting and configuring a primary server in accordance with an embodiment of the present invention. Note that this process is run concurrently by each active node in distributed computing system 100. The system begins by receiving an indication from CMM 210 that the membership in the cluster has changed (step 401).

In response to this indication, the system obtains a lock on a local candidate variable which contains an identifier for a candidate node to host primary server 106 (step 402). The system also obtains an additional lock to hold off requesters for the service (step 404).

Next, the system executes a disqualification process by communicating with other nodes in distributed computing system 100 in order to disqualify the other nodes from acting as the primary server 106 (step 406). This process is described in more detail with reference to FIG. 6 below.

After the disqualification process, the remaining node, which is not disqualified, becomes the primary node. If the node hosting primary server 106 has changed, this may involve re-mapping connections for the service to point to the new node (step 408). It may also involve initializing the new node to act as the host for the primary (step 410).

Finally, the service is started (step 412). This may involve unfreezing the service if it was previously frozen, as well as releasing the previously obtained lock that holds off requesters for the service.

FIG. 5 is a flow chart illustrating some of the operations performed by a primary server 106 in accordance with an embodiment of the present invention. During operation, primary server 106 performs periodic checkpointing operations 120-121 with secondary servers 107-108, respectively (step 502). These checkpointing operations allow secondary servers 107-108 to take over from primary server 106 if primary server 106 fails. Primary server 106 also periodically attempts to promote spare nodes (such as node 105 in FIG. 1) to host secondaries (step 504). This promotion process involves transferring state information to a spare node in order to bring the spare node up to date with respect to the existing secondaries.

FIG. 6 illustrates how a node is selected to host primary server 106 through a disqualification process in accordance with an embodiment of the present invention. Note that FIG. 6 describes in more detail the process described above with reference to step 406 in FIG. 4.

The system starts by determining if a node that was previously hosting primary server 106 continues to exist (step 602).

If not, the system retrieves the state of a local provider for the service (step 604). The system then sets the candidate variable to identify the local provider (step 606), and subsequently unlocks the candidate lock that was set previously in step 402 of FIG. 4 (step 608). Next, if the candidate is not the local provider, the system ends the process (step 610).

Next, for all other nodes I in the cluster, the system attempts to disqualify node I by writing a new identifier into the candidate variable for node I if the rank of node I is less than the rank of the present node (step 612). This process is described in more detail with reference to FIG. 7 below. Finally, if node I's local provider has a higher rank than the present node, the process terminates because the present node is disqualified (step 614).

Note that a rank of a node can be obtained by comparing a unique identifier for the node with unique identifiers for other nodes. Also note that the rank of a primary server is greater than the rank of a secondary server, and that the rank of a secondary server is greater than the rank of a spare. The above-listed restrictions on rank ensure that an existing primary that has not failed continues to function as the primary, and that an existing secondary will be chosen ahead of a spare. Of course, when the system is initialized, no primaries or secondaries exist, so a spare is selected to be the primary.

On the other hand, if the node that was hosting primary server 106 continues to function, the system sets the candidate to be this node (step 616), and unlocks the candidate node (step 618).

If the present node does not host primary server 106, the process is finished. Otherwise, if the present node is hosting primary server 106, the system considers each other node I in the cluster. If the present node has already communicated with I, the system skips node I (step 622). Otherwise, the system communicates with node I in order to disqualify node I from acting as the host for primary server 106 (step 624). This may involve causing an identifier for the present node to be written into the candidate variable for node I.

FIG. 7 illustrates how nodes are disqualified in accordance with an embodiment of the present invention. Note that FIG. 7 describes in more detail the process described above with reference to step 612 in FIG. 6. The caller first locks the candidate variable for node I (step 702). If the caller determines that the caller's provider has a higher rank than is specified in the candidate variable for I, the caller overwrites the candidate variable for I with the caller's provider (step 704). Next, the caller unlocks the candidate variable for I (step 706).

The foregoing descriptions of embodiments of the invention have been presented for purposes of illustration and description only. They are not intended to be exhaustive or to limit the present invention to the forms disclosed. Accordingly, many modifications and variations will be apparent to practitioners skilled in the art. Additionally, the above disclosure is not intended to limit the present invention. The scope of the present invention is defined by the appended claims.

09652553 .091500

What Is Claimed Is:

1. A method for selecting a node to host a primary server for a service from a plurality of nodes in a distributed computing system, the method comprising:

receiving an indication that a state of the distributed computing system has changed;

in response to the indication, determining if there is already a node hosting the primary server for the service; and

if there is not already a node hosting the primary server, selecting a node to host the primary server based upon rank information for the nodes.

2. The method of claim 1, wherein selecting the node to host the primary server involves:

assuming that a given node from the plurality of nodes in the distributed computing system hosts the primary server,

communicating rank information between the given node and other nodes in the distributed computing system, wherein each node in the distributed computing system has a unique rank with respect to the other nodes in the distributed computing system,

comparing a rank of the given node with a rank of the other nodes in the distributed computing system, and

if one of the other nodes in the distributed computing system has a higher rank than the given node, disqualifying the given node from hosting the primary server.

09550553.091500

3. The method of claim 2, further comprising, if there exists a node that is configured to host the primary server, allowing the node that is configured to host the primary server to communicate with other nodes in the distributed computing system in order to disqualify the other nodes from hosting the primary server.

4. The method of claim 2, wherein assuming that the given node hosts the primary server involves:

maintaining a candidate variable in the given node identifying a candidate node to host the primary server; and

initially setting the candidate variable to identify the given node.

5. The method of claim 1, further comprising, after a new node has been selected to host the primary server, if the new node is different from a previous node that hosted the primary server, establishing connections for the service to the new node.

6. The method of claim 1, further comprising, after a new node has been selected to host the primary server, if the new node is different from a previous node that hosted the primary server, configuring the new node to host the primary server for the service.

7. The method of claim 1, further comprising restarting the service if the service was interrupted as a result of the change in state of the distributed computing system.

8. The method of claim 2, wherein the given node in the distributed computing system acts a one of:

a host for the primary server for the service;

a host for a secondary server for the service, wherein the secondary server periodically receives checkpointing information from the primary server; and

a spare for the primary server, wherein the spare does not receive checkpointing information from the primary server.

9. The method of claim 8, further comprising, upon initial startup of the service, selecting a highest ranking spare to host the primary server for the service.

10. The method of claim 8, further comprising allowing the primary server to configure spares in the distributed computing system to host secondary servers for the service.

11. The method of claim 8, wherein comparing the rank of the given node with the rank of the other nodes in the distributed computing system involves considering a host for the primary server to have a higher rank than a host for a space, and considering a host for a secondary server to have a higher rank than a spare.

12. The method of claim 2, wherein disqualifying the given node from hosting the primary server involves ceasing to communicate rank information between the given node and the other nodes in the distributed computing system.

13. A computer-readable storage medium storing instructions that when executed by a computer cause the computer to perform a method for selecting a node to host a primary server for a service from a plurality of nodes in a distributed computing system, the method comprising:

receiving an indication that a state of the distributed computing system has changed;

in response to the indication, determining if there is already a node hosting the primary server for the service; and

if there is not already a node hosting the primary server, selecting a node to host the primary server based upon rank information for the nodes.

14. The computer-readable storage medium of claim 13, wherein selecting the node to host the primary server involves:

assuming that a given node from the plurality of nodes in the distributed computing system hosts the primary server,

communicating rank information between the given node and other nodes in the distributed computing system, wherein each node in the distributed computing system has a unique rank with respect to the other nodes in the distributed computing system,

comparing a rank of the given node with a rank of the other nodes in the distributed computing system, and

if one of the other nodes in the distributed computing system has a higher rank than the given node, disqualifying the given node from hosting the primary server.

15. The computer-readable storage medium of claim 14, wherein if there exists a node that is configured to host the primary server, the method

further comprises allowing the node that is configured to host the primary server to communicate with other nodes in the distributed computing system in order to disqualify the other nodes from hosting the primary server.

16. The computer-readable storage medium of claim 14, wherein assuming that the given node hosts the primary server involves:

maintaining a candidate variable in the given node identifying a candidate node to host the primary server; and

initially setting the candidate variable to identify the given node.

17. The computer-readable storage medium of claim 13, wherein after a new node has been selected to host the primary server, if the new node is different from a previous node that hosted the primary server, the method further comprises establishing connections for the service to the new node.

18. The computer-readable storage medium of claim 13, wherein after a new node has been selected to host the primary server, if the new node is different from a previous node that hosted the primary server, the method further comprises configuring the new node to host the primary server for the service.

19. The computer-readable storage medium of claim 13, wherein the method further comprises restarting the service if the service was interrupted as a result of the change in state of the distributed computing system.

20. The computer-readable storage medium of claim 14, wherein the given node in the distributed computing system acts a one of:

a host for the primary server for the service;

a host for a secondary server for the service, wherein the secondary server periodically receives checkpointing information from the primary server; and

a spare for the primary server, wherein the spare does not receive checkpointing information from the primary server.

21. The computer-readable storage medium of claim 20, wherein upon initial startup of the service, the method further comprises selecting a highest ranking spare to host the primary server for the service.

22. The computer-readable storage medium of claim 20, wherein the method further comprises allowing the primary server to configure spares in the distributed computing system to host secondary servers for the service.

23. The computer-readable storage medium of claim 20, wherein comparing the rank of the given node with the rank of the other nodes in the distributed computing system involves considering a host for the primary server to have a higher rank than a host for a space, and considering a host for a secondary server to have a higher rank than a spare.

24. The computer-readable storage medium of claim 14, wherein disqualifying the given node from hosting the primary server involves ceasing to communicate rank information between the given node and the other nodes in the distributed computing system.

/25. An apparatus that selects a node to host a primary server for a service from a plurality of nodes in a distributed computing system, the apparatus comprising:

a receiving mechanism that is configured to receive an indication that a state of the distributed computing system has changed;

a determination mechanism that is configured to determine if there is already a node hosting the primary server for the service in response to the indication;

a selecting mechanism, wherein if there is not already a node hosting the primary server, the selecting mechanism is configured to select a node to host the primary server based upon rank information for the nodes.

26. The apparatus of claim 25, wherein, in selecting a node to host the primary server based upon rank information, the selecting mechanism is configured to:

communicate rank information between the given node and other nodes in the distributed computing system, wherein each node in the distributed computing system has a unique rank with respect to the other nodes in the distributed computing system, and to

compare a rank of the given node with a rank of the other nodes in the distributed computing system.

27. The apparatus of claim 26, further comprising a disqualification mechanism that is configured to disqualify the given node from hosting the primary server if one of the other nodes in the distributed computing system has a higher rank than the given node.

28. The apparatus of claim 26, further comprising a mechanism on the primary server that is configured to communicate with other nodes in the

distributed computing system in order to disqualify the other nodes from hosting the primary server.

29. The apparatus of claim 26, wherein the selecting mechanism is configured to:

maintain a candidate variable in the given node identifying a candidate node to host the primary server; and to

initially set the candidate variable to identify the given node.

30. The apparatus of claim 25, further comprising a connection mechanism that is configured to establish connections for the service to a new node after the new node has been selected to host the primary server, and if the new node is different from a previous node that hosted the primary server.

31. The apparatus of claim 25, further comprising a mechanism that configures a new node to host the primary server for the service, after the new node has been selected to host the primary server, and if the new node is different from a previous node that hosted the primary server.

32. The apparatus of claim 25, further comprising a restarting mechanism that is configured to restart the service if the service was interrupted as a result of the change in state of the distributed computing system.

33. The apparatus of claim 26, wherein the given node in the distributed computing system acts a one of:

a host for the primary server for the service;

a host for a secondary server for the service, wherein the secondary server periodically receives checkpointing information from the primary server; and

a spare for the primary server, wherein the spare does not receive checkpointing information from the primary server.

34. The apparatus of claim 33, further comprising an initialization mechanism wherein during initialization of the service, the initialization mechanism is configured to select a highest ranking spare to host the primary server for the service.

35. The apparatus of claim 33, further comprising a promotion mechanism on the primary server that that is configured to promote spares in the distributed computing system to host secondary servers for the service.

09652553.081500

36. The apparatus of claim 33, wherein while comparing the rank of the given node with the rank of the other nodes in the distributed computing system, the selecting mechanism is configured to consider a host for the primary server to have a higher rank than a host for a secondary server, and to consider a host for a secondary server to have a higher rank than a spare.

37. The apparatus of claim 26, wherein the selecting mechanism is configured to cease to communicate rank information between the given node and the other nodes in the distributed computing system after the given node is disqualified by the disqualification mechanism.

38. A method for selecting a node to host a primary server for a service from a plurality of nodes in a distributed computer system, comprising:

communicating disqualification information between the node and remaining nodes in the plurality of nodes;

disqualifying the node from hosting the primary server based upon the disqualification information received from the remaining nodes.

39. The method of claim 38, wherein the disqualification information comprises a node rank information.

40. The method of claim 39, wherein the node rank for a given node is calculated using an assumption that the given node hosts the primary server.

41. The method of claim 40, wherein the calculated node rank is unique with respect to the ranks of other nodes in the distributed computer system.

42. The method of claim 39, wherein the disqualifying of the node comprises:

comparing a rank of the node to a set of ranks of the remaining nodes in the distributed computer system; and

disqualifying the node from hosting the primary server if one of the set of ranks of the remaining nodes is higher than the rank of the node.

43. The method of claim 38, further comprising repeating the acts of communicating disqualification information and disqualifying the node for at least one more node in the plurality of nodes.

METHOD AND APPARATUS FOR REACHING AGREEMENT BETWEEN NODES IN A DISTRIBUTED SYSTEM

ABSTRACT

One embodiment of the present invention provides a system for selecting a node to host a primary server for a service from a plurality of nodes in a distributed computing system. The system operates by receiving an indication that a state of the distributed computing system has changed. In response to this indication, the system determines if there is already a node hosting the primary server for the service. If not, the system selects a node to host the primary server using the assumption that a given node from the plurality of nodes in the distributed computing system hosts the primary server. The system then communicates rank information between the given node and other nodes in the distributed computing system, wherein each node in the distributed computing system has a unique rank with respect to the other nodes in the distributed computing system. The system next compares the rank of the given node with the rank of the other nodes in the distributed computing system. If one of the other nodes has a higher rank than the given node, the system disqualifies the given node from hosting the primary server.

09552553.091500

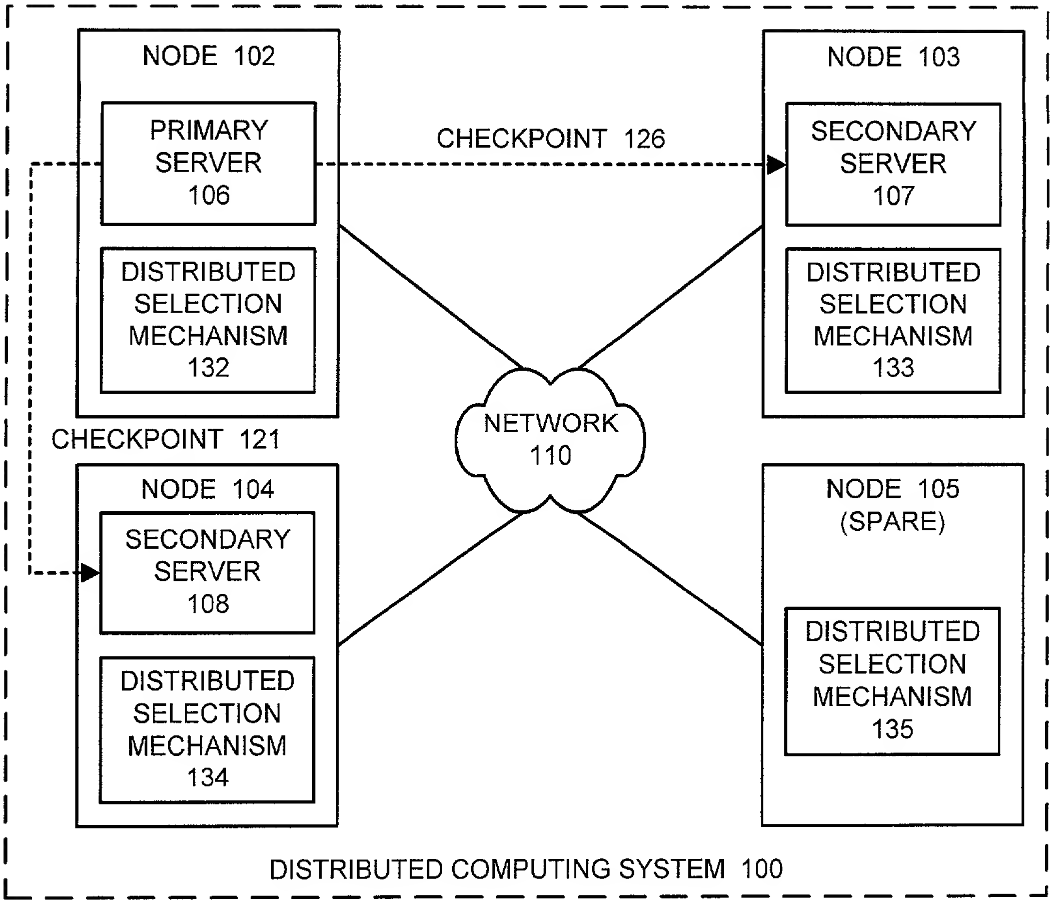

NODE 102
PRIMARY SERVER 106
DISTRIBUTED SELECTION MECHANISM 132
CHECKPOINT 126
NODE 103
SECONDARY SERVER 107
DISTRIBUTED SELECTION MECHANISM 133
NETWORK 110
CHECKPOINT 121
NODE 104
SECONDARY SERVER 108
DISTRIBUTED SELECTION MECHANISM 134
NODE 105 (SPARE)
DISTRIBUTED SELECTION MECHANISM 135
DISTRIBUTED COMPUTING SYSTEM 100

FIG. 1

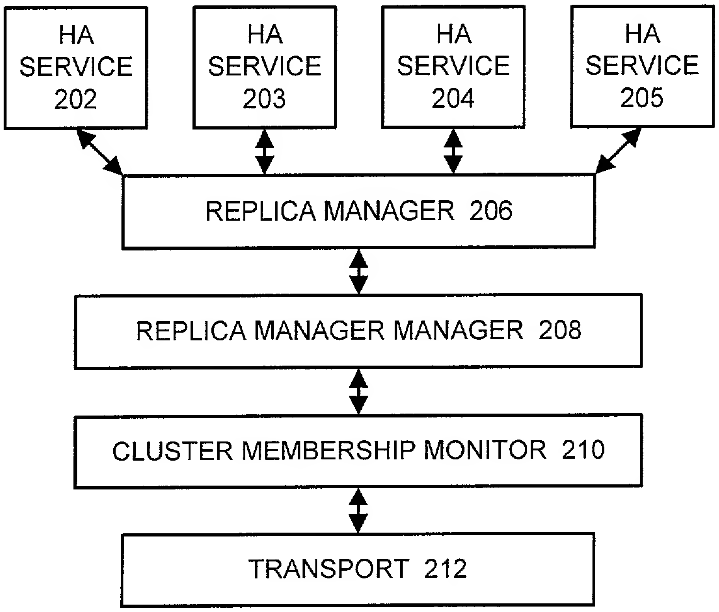
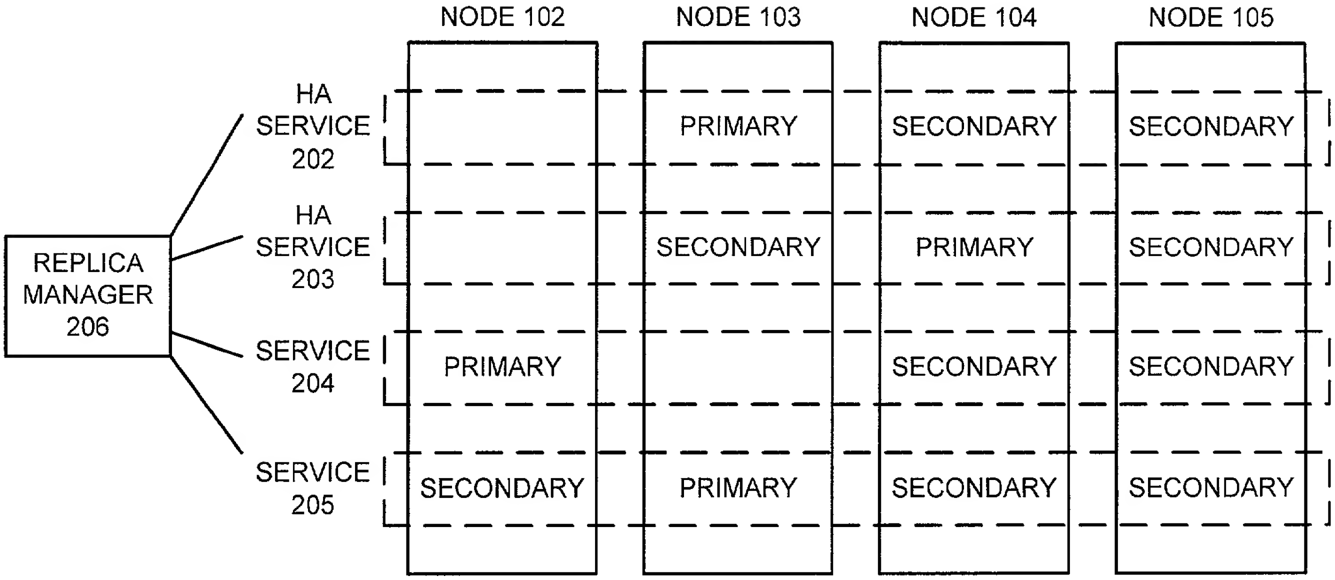

HA SERVICE 202
HA SERVICE 203
HA SERVICE 204
HA SERVICE 205
REPLICA MANAGER 206
REPLICA MANAGER MANAGER 208
CLUSTER MEMBERSHIP MONITOR 210
TRANSPORT 212

FIG. 2



	NODE 102	NODE 103	NODE 104	NODE 105
HA SERVICE 202		PRIMARY	SECONDARY	SECONDARY
HA SERVICE 203		SECONDARY	PRIMARY	SECONDARY
SERVICE 204	PRIMARY		SECONDARY	SECONDARY
SERVICE 205	SECONDARY	PRIMARY	SECONDARY	SECONDARY

FIG. 3

09695253.061500

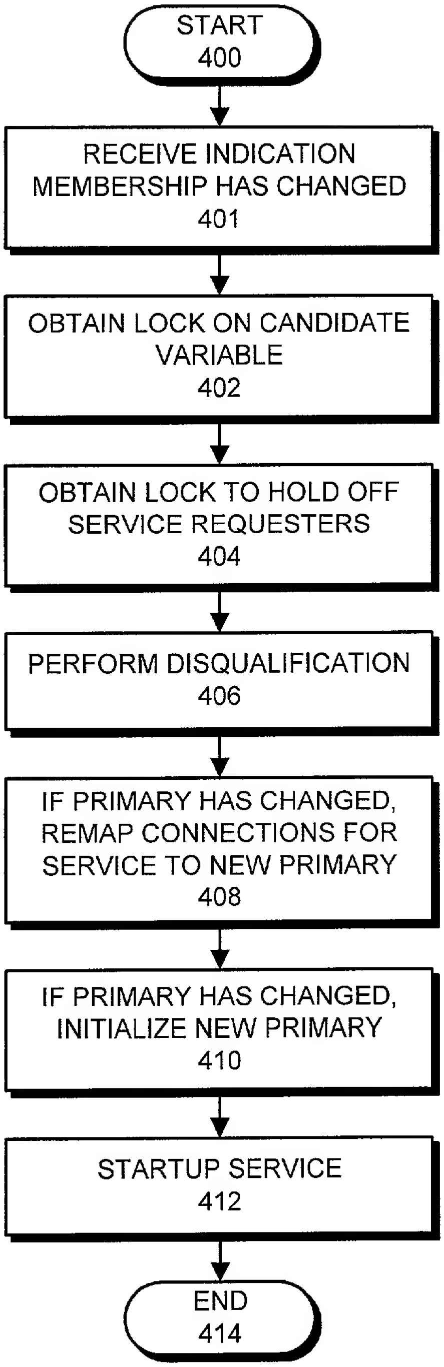

START
400
RECEIVE INDICATION MEMBERSHIP HAS CHANGED
401
OBTAIN LOCK ON CANDIDATE VARIABLE
402
OBTAIN LOCK TO HOLD OFF SERVICE REQUESTERS
404
PERFORM DISQUALIFICATION
406
IF PRIMARY HAS CHANGED, REMAP CONNECTIONS FOR SERVICE TO NEW PRIMARY
408
IF PRIMARY HAS CHANGED, INITIALIZE NEW PRIMARY
410
STARTUP SERVICE
412
END
414

FIG. 4

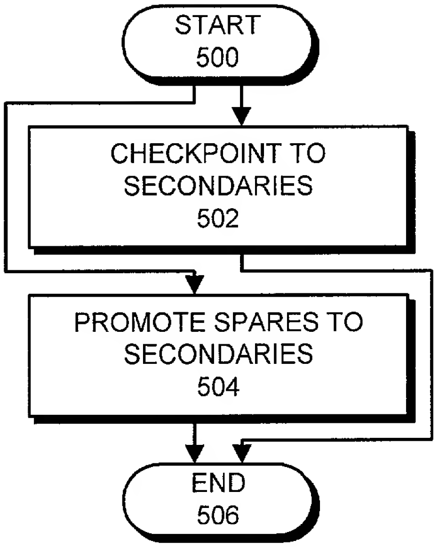


FIG. 5

09652553.091500

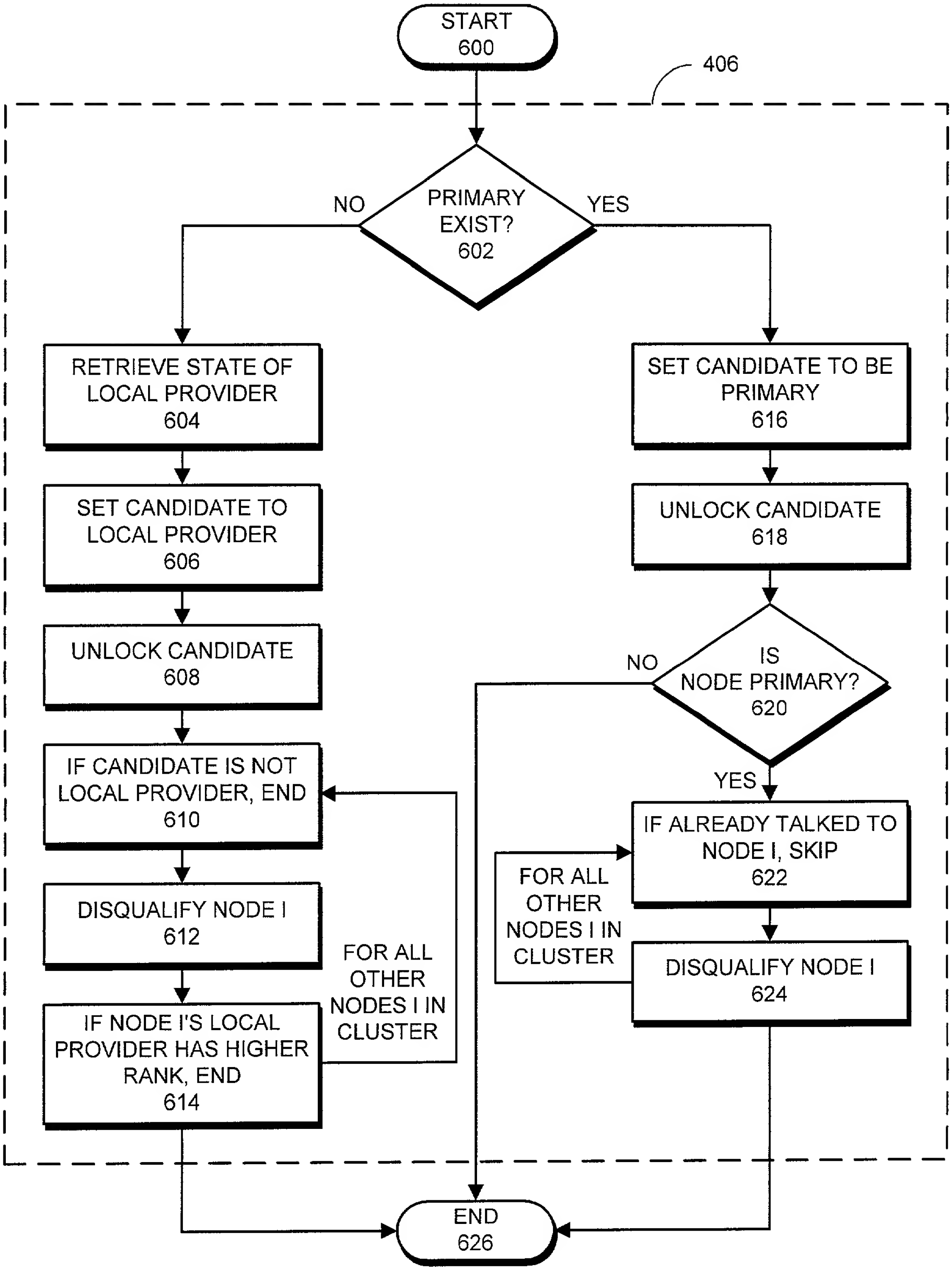

START
600
406
PRIMARY EXIST?
602
NO
YES
RETRIEVE STATE OF LOCAL PROVIDER
604
SET CANDIDATE TO LOCAL PROVIDER
606
UNLOCK CANDIDATE
608
IF CANDIDATE IS NOT LOCAL PROVIDER, END
610
DISQUALIFY NODE I
612
IF NODE I'S LOCAL PROVIDER HAS HIGHER RANK, END
614
FOR ALL OTHER NODES I IN CLUSTER
SET CANDIDATE TO BE PRIMARY
616
UNLOCK CANDIDATE
618
IS NODE PRIMARY?
620
NO
YES
IF ALREADY TALKED TO NODE I, SKIP
622
FOR ALL OTHER NODES I IN CLUSTER
DISQUALIFY NODE I
624
END
626

FIG. 6

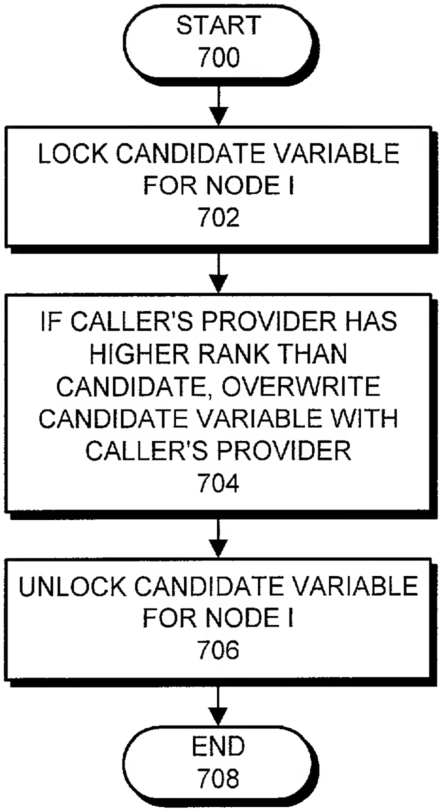

START
700
LOCK CANDIDATE VARIABLE FOR NODE I
702
IF CALLER'S PROVIDER HAS HIGHER RANK THAN CANDIDATE, OVERWRITE CANDIDATE VARIABLE WITH CALLER'S PROVIDER
704
UNLOCK CANDIDATE VARIABLE FOR NODE I
706
END
708

FIG. 7

09625553.091500

Attorney Docket No. SUN-P4431-ARG

COMBINED DECLARATION AND POWER OF ATTORNEY

As a below-named inventor, I hereby declare that:

My residence, post office address and citizenship are as stated below by my name;

I believe I am the original, first and sole inventor, if only one name is listed below, or an original, first and joint inventor if multiple names are listed below, of the subject matter which is claimed and for which a patent is sought on the invention entitled:

METHOD AND APPARATUS FOR REACHING AGREEMENT BETWEEN NODES IN A DISTRIBUTED SYSTEM

for which a patent application:

☒ is attached hereto.

□ was filed in the United States on _ as Application No. ____;
  □ with amendment(s) filed on ________________ *(if applicable)*.

I hereby state that I have reviewed and understand the contents of the application identified above, including the claims, as amended by any amendment referred to above.

I acknowledge the duty to disclose information known to me to be material to the examination of this application in accordance with Title 37, Code of Federal Regulations, §1.56, which states in relevant part:

> Each individual associated with the filing and prosecution of a patent application has a duty of candor and good faith in dealing with the Office, which includes a duty to disclose to the Office all information known to that individual to be material to patentability as defined in this section... The duty to disclose all information known to be material to patentability is deemed to be satisfied if all information known to be material to patentability of any claim issued in a patent was cited by the Office or submitted to the Office...

I hereby claim foreign priority benefits under Title 35, United States Code, §119(a)-(d), of any foreign application(s) for patent or inventor's certificate as indicated below and have also identified below any foreign application for patent or inventor's certificate on this invention having a filing date before that of the application on which priority is claimed:

EARLIEST FOREIGN APPLICATION(S), IF ANY, FILED PRIOR TO THE FILING DATE OF THE APPLICATION			
APPLICATION NUMBER	COUNTRY	DATE OF FILING (Day, Month, Year)	PRIORITY CLAIMED
			YES □ NO □

I hereby claim the benefit under Title 35, United States Code, §119(e), of any United States provisional application(s) listed below:

APPLICATION NUMBER	DATE OF FILING
60/160,992	October 21, 1999

I hereby claim the benefit under Title 35, United States Code, §120, of any United States application(s) listed below and, insofar as the subject matter of each of the claims of this application is not disclosed in the prior United States application in the manner provided by the first paragraph of Title 35, United States Code, §112, I acknowledge the duty to disclose information that is material to patentability as defined in Title 37, Code of Federal Regulations, §1.56, which became available between the filing date of the prior application and the national or PCT international filing date of this application:

APPLICATION NUMBER	DATE OF FILING	STATUS		
		PATENTED	PENDING	ABANDONED

I hereby appoint Daniel E. Vaughan (Reg. No. 42,199) and A. Richard Park (Reg. No. 41,241) to prosecute this application and transact all business in the Patent and Trademark Office connected therewith, and to file, prosecute and transact all business in connection with international applications directed to said invention.

Address correspondence to:
Park & Vaughan LLP
508 Second Street, Suite 201
Davis, CA 95616

22835
PATENT TRADEMARK OFFICE

Direct telephone calls to:
A. Richard Park
(530) 759-1661

I hereby declare that all statements made herein of my own knowledge are true and that all statements made on information and belief are believed to be true; and further that these statements were made with the knowledge that willful false statements and the like so made are punishable by fine or imprisonment, or both, under Title 18, United States Code, §1001, and that such willful false statements may jeopardize the validity of the application or any patent issued thereon.

1	Name and Citizenship	Skef F. Iterum	United States of America
	Residence Address	1290 Rhode Island St. #21, San Francisco, CA 94107	
	Postal Address *(if different from Residence)*		
	Signature and Date		Date 8/24/00
2	Name and Citizenship	Declan J. Murphy	Ireland
	Residence Address	41 Newberg St., San Francisco, CA 94131	
	Postal Address *(if different from Residence)*		
	Signature and Date	Declan Murphy	Date 8/25/00
3	Name and Citizenship		
	Residence Address		
	Postal Address *(if different from Residence)*		
	Signature and Date		Date
4	Name and Citizenship		
	Residence Address		
	Postal Address *(if different from Residence)*		
	Signature and Date		Date
5	Name and Citizenship		
	Residence Address		
	Postal Address *(if different from Residence)*		
	Signature and Date		Date

Additional inventor name(s) and signature(s) attached?: YES ☐ NO ☒

POWER OF ATTORNEY BY ASSIGNEE TO EXCLUSION OF INVENTOR UNDER 37 C.F.R. § 3.71 WITH REVOCATION OF PRIOR POWERS

Inventor(s):	Skef F. Iterum, et al.
Title:	METHOD AND APPARATUS FOR REACHING AGREEMENT BETWEEN NODES IN A DISTRIBUTED SYSTEM
Filing Date:	To Be Assigned
Serial No.:	To Be Assigned
Group Art Unit:	To Be Assigned
Examiner:	To Be Assigned
Attorney Docket No:	SUN-P4431-ARG

The undersigned ASSIGNEE of the entire interest in the above-identified application for letters patent hereby appoints Kenneth Olsen, Reg. No. 26,493, Timothy J. Crean, Reg. No. 37,116, Joseph T. FitzGerald, Reg. No. 33,881, Robert S. Hauser, Reg. No. 37,847, Alexander E. Silverman, Reg. No. 37,940, Christine S. Lam, Reg. No. 37,489, Anirma Rakshpal Gupta, Reg. No. 38,275, Sean P. Lewis, Reg. No. 42,798, Michael J. Schallop, Reg. No. 44,319, Bernice B. Chen, Reg. No. 42,403, Kenta Suzue, Reg. No. 45,145, Noreen A. Krall, Reg. No. 39,734, Richard J. Lutton, Jr., Reg. No. 39,756, Monica D. Lee, Reg. No. 40,696 and Marc D. Foodman, Reg. No. 34,110 all of SUN MICROSYSTEMS, INC., and A. Richard Park, Registration No. 41,241 and Daniel E. Vaughan, Registration No. 42,199 of PARK & VAUGHAN LLP, to prosecute this application and transact all business in the United States and Trademark Office in connection therewith and hereby revokes all prior powers of attorney; said appointment to be to the exclusion of the inventors and the inventors' attorneys in accordance with the provisions of 37 C.F.R. § 3.71.

The following evidentiary documents establish a chain of title from the original owner to the Assignee:

_x_ a copy of an Assignment attached hereto, which Assignment has been (or is herewith) forwarded to the Patent and Trademark Office for recording; or

___ the Assignment recorded on ________ at reel ______, frames ______- ______.

Pursuant to 37 C.F.R.§ 3.73(b) the undersigned Assignee hereby states that evidentiary documents have been reviewed and hereby certifies that, to the best of ASSIGNEE's knowledge and belief, title is in the identified ASSIGNEE.

Please direct all telephone calls and correspondence to: A. Richard Park, Park & Vaughan LLP, 508 Second Street, Suite 201, Davis, CA 95616, tel: (530) 759-1661.

ASSIGNEE: Sun Microsystems, Inc.

Signature: ________________ SEP 12 2000
(Signature) (Date)

Name: Kenneth Olsen

Title: Vice President, Intellectual Property